

Presentation

공유 서비스

KkowBorrow



김도하

김민지

이은서

전현규

조은희

Contents

1

기획

- Tools

- 회의록

- Research

- WBS

2

설계

- 논리적 모델링

- 요구사항 명세서

- 테이블 정의서

- API 명세서

3

구현

- 테스트 케이스

Presentation

기획

Tools

회의록

Research

WBS

Tools



Gradle



MariaDB



Swagger



Spring
Boot



POSTMAN



Java™

Presentation

The screenshot shows the Notion AI interface with a presentation slide. The slide title is '24.07.02 회의록'. The sidebar on the left contains a navigation menu with sections like 'Notion AI' (New), 'Home', 'Inbox', 'Settings & Members', 'Personal Pages', 'Personal Notebooks', '1:1 Meeting Log', 'Start', 'Is this just for me?', 'Team Space', '1st HQ', 'API Notebooks', 'Ability Notebooks', 'Meeting', 'Document', and 'Notion Continues Usage'.

24.07.02 회의록

유형

팀 주간 회의

이벤트 시간

2024년 7월 2일 오후 6:00

참석자

비어 있음

속성 4개 추가

댓글 추가

어젠다

- 요구사항 명세서 확인 및 수정
- ERD 공유 및 수정
- 이후 일정 수립

누르면 안되는 버튼

Presentation

fix: login

feat: logincontroller

docs: Update README.md

fix: merge

Merge branch 'main' of <https://github.com/bey>

refactor: chatcontent

docs: Update README.md

fix: login

docs: Update README.md

Update README.md

Update README.md

Update README.md

Update README.md

docs: Update README.md add test

Update ReservationRequestDto.java

origin & main DESKTOP-QPBSKD8WPlaydata
esueng*

eunhee78*

origin & login DESKTOP-QPBSKD8WPlaydata
eunhee

eunhee

eunhee78*

DESKTOP-QPBSKD8WPlaydata
eunhee78*

tkckdnjs*

tkckdnjs*

tkckdnjs*

tkckdnjs*

eunhee78*

ideapple*

Update ReservationRequestDto.java

Update ReservationRequestDto.java

Update ComplainRequestDto.java

Update README.md

Update README.md

Update README.md

docs: Update README.md

refactor: REST API

docs: Update README.md

docs: Update README.md

ideapple*

ideapple*

ideapple*

tkckdnjs*

tkckdnjs*

tkckdnjs*

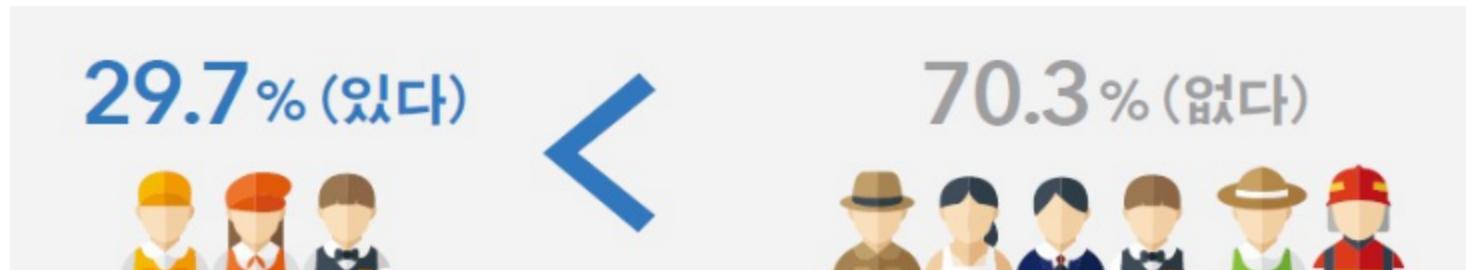
eunhee78*

eunhee

eunhee78*

eunhee78*

Research



공유경제 사용경험에 대한 설문조사에 따르면 **70.3%** 가 서비스를 이용해보지 않았다고 나온다. 이는 공유경제에 대한 인지도 수준이 낮고 국민적 공감대 역시 형성되어 있지 않은 것으로 나타나, 공유경제 시장 환경이 **기대만큼 긍정적이지 않음**을 확인할 수 있다.

이용사유로는 '**편리성**'이 **40.1%**로 가장 높았다('저렴한 비용' 29.0%). 한편, 공유경제 서비스 이용자의 65.3%가 서비스에 만족한다고 응답해 **이용 만족도는 대체로 높은 편**이었다.



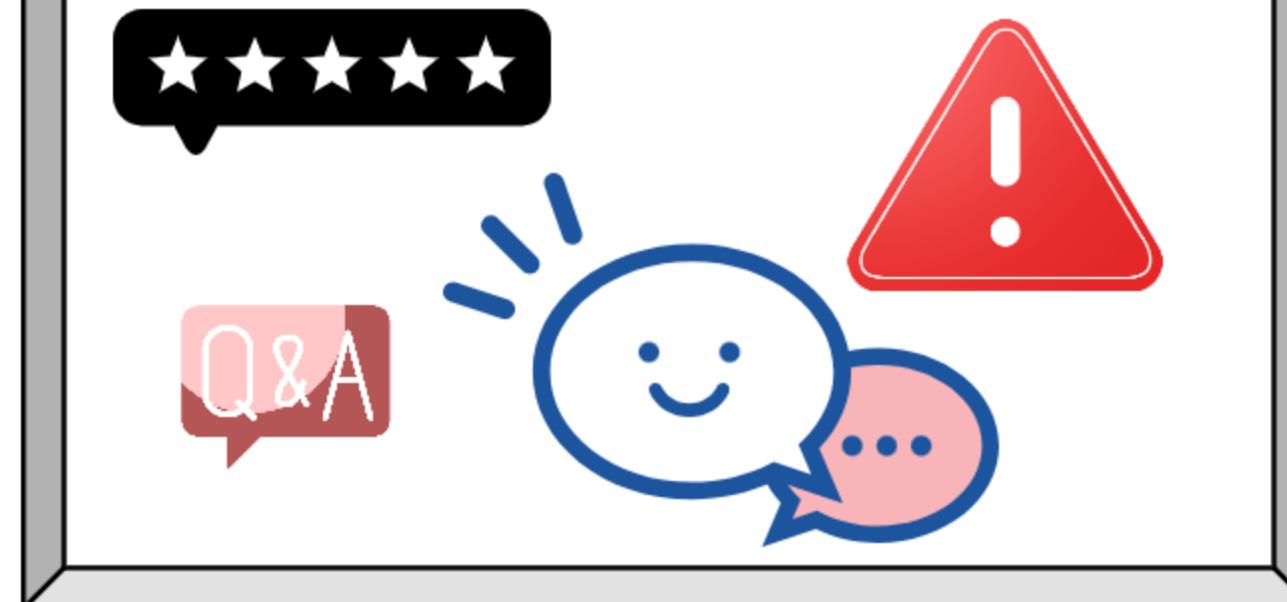
꿔 Borrow

1 대여 / 반납 서비스



개인이 모든 자원을 소유할 필요 없이 필요한 시간에 대여하여 사용할 수 있기 때문에, 불필요한 소비를 멈출 수 있다.

2 게시판 기능



문의사항, 채팅, 리뷰, 신고 기능들을 통해 사용자 경험을 향상시킬수 있다.

Presentation

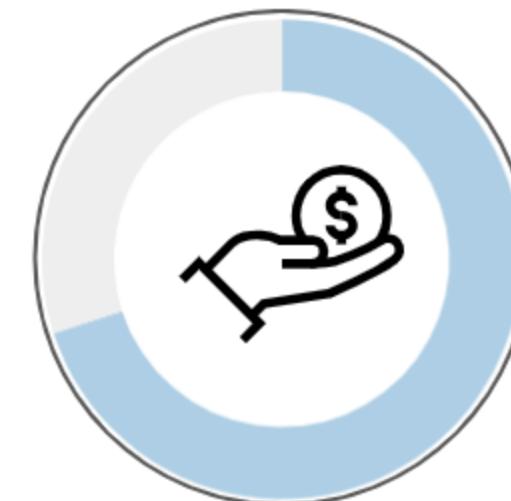
借用 Borrow



기대효과

경제 활성화

물건을 빌려주는 사람은 추가 수익을 얻고, 빌리는 사람은 구매 비용을 절약 할 수 있다.



환경 보호

불필요한 소비를 줄이고 환경 보호에 기여할 수 있다.



접근성 향상

다양한 자원에 대한 접근성이 증대되어 경제적 약자도 고가의 자산이나 서비스를 이용할 수 있다.



커뮤니티 강화

효율적인 자원 사용과 지속 가능한 소비 문화를 촉진하는데 큰 역할을 할 수 있다.



Presentation

WBS

설계

논리적 모델링

요구사항 명세서

테이블 정의서

API 명세서

Presentation

논리적 모델링



Presentation

요구사항 명세서

요구사항명				요구사항 내용	완료여부	담당자
대분류	중분류	소분류	요구사항-ID			
회원	회원 관리	회원 가입	REQ01	사용자가 시스템에 새로운 계정을 등록할 수 있도록 한다.	0	현규
		로그인	REQ02	사용자가 시스템에 접근할 수 있도록 인증한다.	0	현규
		로그아웃	REQ03	사용자가 시스템에서 안전하게 로그아웃할 수 있도록 한다.	0	현규
		회원 탈퇴	REQ04	사용자가 시스템에서 자신의 계정을 영구적으로 삭제할 수 있도록 한다.	0	현규
	마이페이지	회원 정보 수정	REQ05	사용자가 자신의 개인 정보를 수정할 수 있도록 한다.	0	현규
		내 등록 물품 조회	REQ06	사용자가 자신이 등록한 물품을 조회할 수 있도록 한다.	0	현규
		내 반납 물품 조회	REQ07	사용자가 반납한 물품을 조회할 수 있도록 한다.	0	현규
		내 대여 상황 조회	REQ08	사용자가 현재 대여 중인 물품을 조회할 수 있도록 한다.	0	현규
		내 문의 사항 조회	REQ09	사용자가 자신의 문의 사항을 조회할 수 있도록 한다.	0	현규
	채팅	회원 정보 조회	REQ10	사용자가 자신의 회원 정보를 조회할 수 있도록 한다.	0	현규
		채팅 보내기	REQ11	사용자가 다른 사용자와 메시지를 주고받을 수 있도록 한다.	0	은희
		채팅 목록 조회	REQ12	사용자가 최근 채팅 목록을 조회할 수 있도록 한다.	0	은희
		채팅 나가기	REQ13	사용자가 특정 채팅방에서 나갈 수 있도록 한다.	0	은희
		채팅 신고	REQ14	사용자가 부적절한 채팅을 신고할 수 있도록 한다.	0	은희
물품	물품 대여	물품 대여	REQ15	사용자가 물품을 대여할 수 있도록 한다.	0	도하
		물품 예약 대기	REQ16	사용자가 대여할 물품을 예약할 수 있도록 한다.	0	도하
	물품 반납	물품 반납	REQ17	사용자가 대여한 물품을 반납할 수 있도록 한다.	0	도하
		회원 멀접 투여	REQ18	사용자가 다른 회원에게 멀접을 부여할 수 있도록 한다.	0	도하
	물품 검색	물품 이름으로 검색	REQ19	사용자가 물품 이름으로 검색할 수 있도록 한다.	0	도하
		카테고리별 물품 검색	REQ20	사용자가 특정 카테고리의 물품을 검색할 수 있도록 한다.	0	도하
		최근 등록된 물품 조회	REQ21	사용자가 최근에 등록된 물품을 조회할 수 있도록 한다.	0	도하
		대여 가능한 물품 조회	REQ22	사용자가 현재 대여 가능한 물품을 조회할 수 있도록 한다.	0	도하
		지역별 물품 검색	REQ23	사용자가 특정 지역의 물품을 검색할 수 있도록 한다.	0	도하
게시판	물품	게시글 등록	REQ24	사용자는 게시글을 등록할 수 있다.	0	은서
		게시글 조회	REQ25	사용자는 다른 게시글을 조회할 수 있다.	0	은서
		게시글 정보 삭제	REQ26	사용자는 작성한 게시글을 삭제할 수 있다.	0	은서
		게시글 정보 수정	REQ27	사용자는 작성한 게시글의 정보를 수정할 수 있다.	0	은서
	문의	게시글 신고	REQ28	사용자는 게시글을 신고할 수 있다.	0	은서
		문의사항 등록	REQ29	사용자가 시스템에 문의사항을 등록할 수 있도록 한다.	0	은서
		문의사항 수정	REQ30	사용자가 자신의 문의사항을 수정할 수 있도록 한다.	0	은서
		문의사항 삭제	REQ31	사용자가 자신의 문의사항을 삭제할 수 있도록 한다.	0	은서
		문의사항 조회	REQ32	사용자가 자신의 문의사항을 조회할 수 있도록 한다.	0	은서
		대여 일정	REQ33	대여 시간이 다가오면 당사자에게 시간, 장소 등의 정보를 담은 알림을 발송한다.	0	민지
알림	알림	반납 일정	REQ34	반납 시간이 다가오면 당사자에게 시간, 장소 등의 정보를 담은 알림을 발송한다.	0	민지
		대여 기능 알림	REQ35	대여 예약을 걸어 놓은 물품이 대여 가능해지면 예약을 걸어놓은 회원들에게 대여가 가능하다는 알림을 발송한다.	0	민지
		후기	후기 작성	대여/반납이 완료된 건에 대해 후기를 작성할 수 있다.	0	민지
	관리자	후기 수정	REQ36	본인이 작성한 후기에 대해 수정이 가능하다.	0	민지
		후기 삭제	REQ37	본인이 작성한 후기에 대해 삭제가 가능하다.	0	민지
	계시판 관리	후기 조회	REQ38	물품을 이용한 회원들의 후기를 조회할 수 있다.	0	민지
		신고 내역 조회	REQ39	관리자가 신고 내역을 조회할 수 있도록 한다.	0	은희
		회원 계정 삭제	REQ40	관리자가 특정 회원의 계정을 삭제할 수 있도록 한다.	0	은희
		게시글 삭제	REQ41	관리자가 특정 게시글을 삭제할 수 있도록 한다.	0	은희
		채팅 삭제	REQ42	관리자가 부적절한 채팅을 삭제할 수 있도록 한다.	0	은희
		Q & A	REQ43	관리자가 부적절한 채팅을 삭제할 수 있도록 한다.	0	은희
		공지사항 등록	REQ44	관리자는 사용자가 등록한 문의에 답변을 남길 수 있도록 한다.	0	은희
		공지사항 수정	REQ45	관리자가 등록된 공지사항을 수정할 수 있도록 한다.	0	은희
		공지사항 삭제	REQ46	관리자가 등록된 공지사항을 삭제할 수 있도록 한다.	0	은희
		FAQ 등록	REQ47	관리자가 등록된 FAQ를 등록할 수 있도록 한다.	0	은희

Presentation

테이블 정의서

테이블정의서			시스템명	쿼리Borrow	작성일	2024/07/07
테이블명	Users	한글명	회원	작성자	김도하	
테이블정의	회원 정보 저장 테이블					
No	컬럼명	컬럼한글명	데이터타입	Null PK	컬럼정의	비고
1	user_id	유저 아이디	LONG	X O	유저 아이디	
2	user_name	유저명	VARCHAR(255)X		유저 성명	
3	password	유저 비밀번호	VARCHAR(255)X		유저 비밀번호	
4	email	유저 이메일	VARCHAR(255)		유저 이메일 주소	
5	user_rate	유저 평점	DECIMAL(1,2)X		기재 평점	
6	user_type	유저 분류	ENUM X		유저 종류 (사용자, 관리자)	
7	address	유저 주소	VARCHAR(255)X		유저 위치 주소	
8	rate_count	유저 평점 수	INT		유저 평점 수	

테이블정의서			시스템명	쿼리Borrow	작성일	2024/07/07
테이블명	Rental	한글명	대여&반납	작성자	김도하	
테이블정의	대여반납 정보 저장 테이블					
No	컬럼명	컬럼한글명	데이터타입	Null PK	컬럼정의	비고
1	transaction_id	대여반납 아이디	LONG	X O	대여반납 아이디	
2	rental_date	대여일자	DATETIME	X	대여일자	
3	return_date	반납일자	DATETIME	X	반납일자	
4	is_return	반납여부	BOOLEAN	X	반납여부	
5	post_id	게시글 아이디	INT	X	게시글 아이디	FK

테이블정의서			시스템명	쿼리Borrow	작성일	2024/07/07
테이블명	Posts	한글명	게시글	작성자	김도하	
테이블정의	게시글 정보 테이블					
No	컬럼명	컬럼한글명	데이터타입	Null PK	컬럼정의	비고
1	post_id	게시글 아이디	LONG	X O	게시글 아이디	
2	title	제목	VARCHAR(255)		게시글 제목	
3	post_content	내용	VARCHAR(255)		게시글 내용	
4	upload_date	업로드 일자	DATETIME	X	게시글 업로드 일자	
5	update_date	업데이트 일자	DATETIME		게시글 최근 업데이트 일자	
6	user_id	유저 아이디	INT	X	유저 아이디	FK
7	product_id	제품 아이디	INT	X	제품 아이디	FK

테이블정의서			시스템명	쿼리Borrow	작성일	2024/07/07
테이블명	Products	한글명	제품	작성자	김도하	
테이블정의	제품 정보 저장 테이블					
No	컬럼명	컬럼한글명	데이터타입	Null PK	컬럼정의	비고
1	product_id	제품 아이디	LONG	X O	제품 아이디	
2	product_name	제품명	VARCHAR(255)X		제품명	
3	location	지역	VARCHAR(255)X		제품 대여 지역	
4	available	대여 가능 여부	BOOLEAN	X	현재 대여 가능 여부	
5	user_id	유저 아이디	INT	X	제품 대여 유저 아이디	FK
6	category	카테고리 아이디	ENUM	X	용품, 전자제품, 생활용품, 쿠미용품, 생활가전	FK
7	price	가격	INT	X	가격	

Presentation

API 명세서

API 명세서 ...

Index		Aa 기능	Http	API Path	담당자	개발 현황	+	...
[회원 가입]	회원가입				김도하	개발 완료		
	로그인				김도하	개발 완료		
	로그아웃				김도하	개발 완료		
[알람]	알람 삭제	DELETE	/api/v1/alarm-service/alarms/{alarm-id}	전현규	개발 완료			
	알람 목록 조회	GET	/api/v1/alarm-service/alarms	전현규	개발 완료			
	특정 알람 조회	GET	/api/v1/alarm-service/alarms/{alarm-id}	전현규	개발 완료			
	알람 정보 수정	PATCH	/api/v1/alarm-service/alarms/{alarm-id}	전현규	개발 완료			
	알람 추가	POST	/api/v1/alarm-service/alarms	전현규	개발 완료			
[신고]	신고 삭제	DELETE	/api/v1/complain-service/complains/{complaintId}	전현규	개발 완료			
	신고 목록 조회	GET	/api/v1/complain-service/complains	전현규	개발 완료			
	신고 조회	GET	/api/v1/complain-service/complains/{complaintId}	전현규	개발 완료			
	신고 생성	POST	/api/v1/complain-service/complains	전현규	개발 완료			
[대여 / 반납]	대여 신청	POST	/api/v1/rental-service/{postId}	이은서	개발 완료			
	반납 신청	POST	/api/v1/rental-	이은서	개발 완료			

[회원]	회원 정보 삭제	PATCH	/api/v1/user-service/users/{user-id}	김민지	개발 완료
특정 회원 조회	GET	/api/v1/user-service/users/{user-id}	김민지	개발 완료	
회원 정보 수정	PATCH	/api/v1/user-service/users/{user-id}	김민지	개발 완료	
회원 저장	POST	/api/v1/user-service/users	김민지	개발 완료	
[채팅 목록]	특정 채팅 목록 조회	GET	/api/v1/chatlist-service/chat/{chat-id}	조은희	개발 완료
	사용자별 채팅 목록 조회	GET	/api/v1/chatlist-service/chats/user/{user-id}	조은희	개발 완료
채팅 목록 생성	POST	/api/v1/chatlist-service/chat	조은희	개발 완료	
채팅 목록 삭제	DELETE	/api/v1/chatlist-service/chat/{chat-id}	조은희	개발 완료	
특정 사용자의 모든 채팅 목록 삭제	DELETE	/api/v1/chatlist-service/chats/user/{user-id}	조은희	개발 완료	
[채팅/문의사항]	문의사항 내용 등록	POST	/api/v1/content-service/content/noti	조은희	개발 완료
	채팅 내용 생성	POST	/api/v1/content-service/content/chat	조은희	개발 완료
[문의사항]	문의사항 내용 생성	POST	/api/v1/content-service/content/noti	조은희	개발 완료
	채팅/문의사항 내용 수정	PATCH	/api/v1/content-service/content/chat/{content-id}	조은희	개발 완료
특정 채팅/문의사항 내용 조회	GET	/api/v1/content-service/content/content/{content-id}/{user-id}	조은희	개발 완료	
특정 채팅/문의사항 내용 삭제	DELETE	/api/v1/content-service/content/chat/{content-id}/{user-id}	조은희	개발 완료	
모든 채팅/문의사항 내용 조회	GET	/api/v1/content-service/contents/{user-id}	조은희	개발 완료	
모든 채팅/문의사항 내용 삭제	DELETE	/api/v1/content-service/contents/{user-id}	조은희	개발 완료	

Presentation

구현

테스트 케이스

Presentation

Alarms APIs 알람 관련 API 목록



Complains APIs 신고 관련 API 목록



Rental APIs 렌탈 관련 API 목록



Products APIs 제품 관련 API 목록



ChatList APIs 채팅 목록 관련 API 목록



Notification APIs 문의사항 목록 관련 API 목록



Users APIs 회원 관련 API 목록



ChatContent APIs 채팅 내용 관련 API 목록



review-controller



post-controller



BackEnd

테스트 케이스 | 회원가입

Register

Username: 김도하

Password:

Email: kdh@beyond.com

Address: dongjak

Register

Login

Username:

Password:

Login

Register

Presentation

테스트 케이스 | 로그인

Welcome, 김도하!

Your email: kdh@beyond.com

Your address: dongjak

[Go to User Info](#)

[View Posts](#)

[Logout](#)

Presentation

테스트 케이스 | 회원 저장

POST /api/v1/user-service/users 회원 저장

회원을 저장한다.

Parameters

No parameters

Request body required

application/json

Example Value | Schema

```
{  
  "user_name": "beyond",  
  "email": "beyond@naver.com",  
  "password": "1234",  
  "user_rate": 0,  
  "user_type": "USER",  
  "address": "string",  
  "rate_count": 0  
}
```

Responses

Code	Description	Links
200	OK Media type /* Controls Accept header.	No links

Presentation

테스트 케이스 | 회원 수정

PATCH /api/v1/user-service/users/{user-id} 회원 정보 수정

특정 회원의 정보를 수정한다.

Parameters

Try it out

Name	Description
user-id <small>* required</small>	회원 번호 <i>Example : 1</i> integer(\$int64) (path)

Responses

Code	Description	Links
200	OK	<i>No links</i>

Media type

`*/*` ▼

Controls Accept header.

[Example Value](#) | [Schema](#)

```
{  
  "userName": "string",  
  "email": "string",  
  "userRate": 0,  
  "userType": "USER",  
  "address": "string"  
}
```

Presentation

테스트 케이스 | 회원 조회

GET /api/v1/user-service/users/{user-id} 특정 회원 조회

특정 회원의 정보를 조회한다.

Cancel

Parameters

Name	Description
user-id * required	회원 번호 integer(\$int64) (path)

Server response

Code Details

200 Response body

```
{  
    "userName": "beyond",  
    "email": "beyond",  
    "userRate": 0,  
    "userType": "USER",  
    "address": "서울특별시"  
}
```

Download

Response headers

```
connection: keep-alive  
content-type: application/json  
date: Tue, 30 Jul 2024 15:50:37 GMT  
keep-alive: timeout=60  
transfer-encoding: chunked
```

Presentation

테스트 케이스 | 물품 대여

POST /api/v1/rental-service/{postId} 대여 신청

대여 가능한 물품의 목록을 조회한다.

Cancel

Parameters

Name	Description
postId <small>* required</small>	integer(\$int64) (path)

1

Responses

Curl

```
curl -X 'POST' \
  'http://localhost:8080/api/v1/rental-service/1' \
  -H 'accept: */*' \
  -d ''
```



Request URL

```
http://localhost:8080/api/v1/rental-service/1
```

Server response

Code Details

200

Response body

```
{
  "postId": 1,
  "transactionID": 1,
  "isReturn": true,
  "rentalDate": "2023-01-10T09:00:00",
  "returnDate": "2023-01-20T09:00:00"
}
```



Download

Presentation

테스트 케이스 | 물품 반납

POST /api/v1/rental-service/return/{transactionID} 반납 신청

반납해야 할 물품 목록을 조회한다.

Parameters

Name	Description
transactionID * required	integer(\$int64) (path)

Cancel

1

Responses

Curl

```
curl -X 'POST' \
'http://localhost:8080/api/v1/rental-service/return/1' \
-H 'accept: */*' \
-d ''
```



Request URL

<http://localhost:8080/api/v1/rental-service/return/1>

Server response

Code Details

200

Response body

```
{  
  "postId": 1,  
  "transactionID": 1,  
  "isReturn": true,  
  "rentalDate": "2023-01-10T09:00:00",  
  "returnDate": "2023-01-20T09:00:00"  
}
```



Download

Presentation

테스트 케이스 | 알람 조회

GET /api/v1/alarm-service/alarms 알람 목록 조회

전체 알람의 목록을 조회한다.

Cancel

Parameters

Name	Description
page * required integer(\$int32) (query)	페이지 번호 페이지 번호
numOfRows * required integer(\$int32) (query)	한 페이지 결과 수 한 페이지 결과 수

Code Details

200 Response body

```
{
  "code": 200,
  "message": "OK",
  "result": [
    {
      "alarmId": 1,
      "meetPlace": "seoul",
      "meetTime": "2024-07-31T00:57:05.272",
      "alarmType": "Reservation",
      "transactionId": null,
      "reservationId": 1
    },
    {
      "alarmId": 2,
      "meetPlace": null,
      "meetTime": "2023-04-01T10:30:00",
      "alarmType": "Reservation",
      "transactionId": null,
      "reservationId": 1
    },
    {
      "alarmId": 3,
      "meetPlace": "Photography Studio C",
      "meetTime": null,
      "alarmType": "Rental",
      "transactionId": 2,
      "reservationId": null
    }
  ]
}
```

Download

Response headers

```
connection: keep-alive
content-type: application/json
date: Wed, 31 Jul 2024 01:03:59 GMT
keep-alive: timeout=60
transfer-encoding: chunked
```

Responses

테스트 케이스 | 알람 삭제

DELETE /api/v1/alarm-service/alarms/{alarm-id} 알람 삭제 ^

특정 알람을 삭제한다.

Cancel

Parameters

Name	Description
alarm-id * required	알람 ID integer(\$int64) (path)
1	

Responses

Curl

```
curl -X 'DELETE' \
'http://localhost:8080/api/v1/alarm-service/alarms/1' \
-H 'accept: application/json'
```



Request URL

```
http://localhost:8080/api/v1/alarm-service/alarms/1
```

Server response

Presentation

테스트 케이스 | 알람 정보 수정

PATCH /api/v1/alarm-service/alarms/{alarm-id} 알람 정보 수정

특정 알람의 정보를 수정한다.

Cancel Reset

Parameters

Name	Description
alarm-id <small>* required</small>	알람 ID integer(\$int64) (path)

Request body required

application/json

```
{  
  "meetPlace": "seoul",  
  "meetTime": "2024-07-31T00:57:05.272Z",  
  "alarmType": "Reservation",  
  "transactionId": 0,  
  "reservationId": 1  
}
```

Responses

Curl

```
curl -X 'PATCH' \  
  'http://localhost:8080/api/v1/alarm-service/alarms/1' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "meetPlace": "seoul",  
    "meetTime": "2024-07-31T00:57:05.272Z",  
    "alarmType": "Reservation",  
    "transactionId": 0,  
    "reservationId": 1  
}'
```

Request URL

```
http://localhost:8080/api/v1/alarm-service/alarms/1
```

Server response

Code Details

200

Response body

```
{  
  "alarmId": 1,  
  "meetPlace": "seoul",  
  "meetTime": "2024-07-31T00:57:05.272Z",  
  "alarmType": "Reservation",  
  "transactionId": null,  
  "reservationId": 1  
}
```



Download

Presentation

테스트 케이스 | 알람 추가

POST /api/v1/alarm-service/alarms 알람 추가

새로운 알람을 추가한다.

Parameters

No parameters

Cancel Reset

Request body required

application/json

```
{  
  "meetPlace": "seoul",  
  "meetTime": "2024-07-31T01:05:09.284Z",  
  "alarmType": "Reservation",  
  "transactionId": 0,  
  "reservationId": 1  
}
```

Responses

Curl

```
curl -X 'POST' \  
  'http://localhost:8080/api/v1/alarm-service/alarms' \  
  -H 'accept: application/json' \  
  -H 'Content-Type: application/json' \  
  -d '{  
    "meetPlace": "seoul",  
    "meetTime": "2024-07-31T01:05:09.284Z",  
    "alarmType": "Reservation",  
    "transactionId": 0,  
    "reservationId": 1  
}'
```

Request URL

http://localhost:8080/api/v1/alarm-service/alarms

Server response

Code Details

200

Response body

```
{  
  "alarmId": 11,  
  "meetPlace": "seoul",  
  "meetTime": "2024-07-31T01:05:09.284",  
  "alarmType": "Reservation",  
  "transactionId": null,  
  "reservationId": 1  
}
```

Download

Presentation

테스트 케이스 | 채팅 내용 등록

POST /api/v1/content-service/content/chat 채팅 내용 등록

새로운 채팅 내용을 등록한다.

Parameters

No parameters

Request body required

application/json

Example Value | Schema

```
{  
  "detail": "테스트 채팅/문의사항 내용입니다.",  
  "userId": 1  
}
```

Server response

Code Details

201 Response body

```
{  
  "contentId": 1,  
  "detail": "테스트 채팅/문의사항 내용입니다.",  
  "sendTime": "2024-07-31T10:20:43.3526065",  
  "userId": 1,  
  "chatId": 1,  
  "notificationId": null  
}
```

Download

Response headers

```
connection: keep-alive  
content-type: application/json  
date: Wed, 31 Jul 2024 01:20:43 GMT  
keep-alive: timeout=60  
transfer-encoding: chunked
```

Presentation

테스트 케이스 | 문의사항 내용 등록

POST /api/v1/content-service/content/noti 문의사항 내용 등록

새로운 문의사항을 등록한다.

Parameters

No parameters

Request body required

application/json

Example Value | Schema

```
{  
  "detail": "테스트 채팅/문의사항 내용입니다.",  
  "userId": 1  
}
```

Try it out Reset

Server response

Code Details

201

Response body

```
{  
  "contentId": 1,  
  "detail": "테스트 채팅/문의사항 내용입니다.",  
  "sendTime": "2024-07-31T10:24:16.7295699",  
  "userId": 1,  
  "chatId": null,  
  "notificationId": 1  
}
```

Download

Response headers

```
connection: keep-alive  
content-type: application/json  
date: Wed, 31 Jul 2024 01:24:16 GMT  
keep-alive: timeout=60  
transfer-encoding: chunked
```

Presentation

테스트 케이스 | 특정 채팅/문의사항 내용 수정

PATCH /api/v1/content-service/content/chat/{content-id}

특정 채팅/문의사항 내용을 수정한다.

Parameters

Name	Description
content-id * required	채팅/문의사항 내용 ID Example : 1
integer(\$int64) (path)	1

Try it out Reset

Request body **required**

application/json

Example Value | Schema

```
{  
  "detail": "테스트 문의사항 내용입니다 !",  
  "userId": 1  
}
```

Server response

Code Details

200

Response body

```
{  
  "contentId": 1,  
  "detail": "테스트 문의사항 내용입니다 !",  
  "sendTime": "2024-07-31T10:29:03.315109",  
  "userId": 1,  
  "chatId": null,  
  "notificationId": 1  
}
```

Download

Response headers

```
connection: keep-alive  
content-type: application/json  
date: Wed, 31 Jul 2024 01:29:03 GMT  
keep-alive: timeout=60  
transfer-encoding: chunked
```

Presentation

thank you!

여기까지 읽어주셔서 감사합니다.



.borrow

BEYOND_SW 8기

김도하 김민지 이은서 전현규 조은희

2024.06.24 ~ 2024.07.31

